# A Framework of Covariance Projection on Constraint Manifold for Data Fusion †

**Muhammad Abu Bakr and Sukhan Lee \***

Intelligent Systems Research Institute, Sungkyunkwan University, Suwon, Gyeonggi-do 440-746, Korea; abubakr@skku.edu

**\*** Correspondence: lsh1@skku.edu; Tel.: +82-31-299-6471

† This paper is an extended version of the paper entitled "A general framework for data fusion and outlier removal in distributed sensor networks", presented at IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Daegu, Korea, 16–18 November 2017.

**Abstract:** A general framework of data fusion is presented based on projecting the probability distribution of true states and measurements around the predicted states and actual measurements onto the constraint manifold. The constraint manifold represents the constraints to be satisfied among true states and measurements, which is defined in the extended space with all the redundant sources of data such as state predictions and measurements considered as independent variables. By the general framework, we mean that it is able to fuse any correlated data sources while directly incorporating constraints and identifying inconsistent data without any prior information. The proposed method, referred to here as the Covariance Projection (CP) method, provides an unbiased and optimal solution in the sense of minimum mean square error (MMSE), if the projection is based on the minimum weighted distance on the constraint manifold. The proposed method not only offers a generalization of the conventional formula for handling constraints and data inconsistency, but also provides a new insight into data fusion in terms of a geometric-algebraic point of view. Simulation results are provided to show the effectiveness of the proposed method in handling constraints and data inconsistency.

**Keywords:** Bar-Shalom Campo; Covariance Projection method; data fusion; distributed architecture; Kalman filter; linear constraints; inconsistent data

## 1. Introduction

Data fusion is the process of obtaining a more meaningful and precise estimate of a state by combining data from multiple sources. The architecture of multisensor data fusion can be broadly categorized into two, depending on the way raw data are processed: (1) Centralized fusion architecture [1], where raw data from multiple sources are directly sent to and fused in the central node for state estimation and (2) Distributed fusion architecture [1,2], where data measured at multiple sources are processed independently at individual nodes to obtain local estimates before they are sent to the central node for fusion. In the centralized architecture, it is possible to apply data fusion methodology such as the Kalman filter [3] to all raw data received to yield optimal estimates in the sense of minimum variance. However, the centralized architecture can be costly especially for a large system in terms of infrastructure and communication overheads at the central node, let alone the issues of reliability and scalability. On the other hand, the distributed architecture is advantageous in reliability and scalability, with lower infrastructure and communication costs. Although advantageous, the distributed architecture needs to address statistical dependency among the local state estimates received from multiple nodes for fusion. This is due to the fact that local state

estimates at individual nodes can be subject to the same process noise [4] and to double counting, i.e., sharing the same data sources among them [5]. Ignoring such statistical dependency or cross-correlation among multiple nodes leads to inconsistent results, causing divergence in data fusion [6]. In the case of known cross-correlation, the Bar-Shalom Campo (BC) formula [7] provides a consistent fusion result for a pair of redundant data sources, where the fused estimate is based on maximum likelihood [8]. A generalization to more than two data sources with known cross-correlations is given by weighted fusion algorithms of the generalized Millman's formula (GMF) [9] and weighted Kalman filter (WKF) [10].

Sensors often provide spurious and inconsistent data due to unexpected situations such as short duration spike faults, sensor glitches, a permanent failure or slowly developing failure due to sensor elements [5,11]. Since these types of uncertainties are not attributable to the inherent noise, they are difficult to predict and model. The fusion of inconsistent sensor data with correct data can lead to severely inaccurate results [12]. For example, when exposed to abnormalities and outliers, a Kalman filter would easily diverge [13]. Hence, a data validation scheme is required to identify and eliminate the sensor faults/outliers/inconsistencies before fusion.

The detection of inconsistency needs either a priori information often in the form of specific failure model(s) or data redundancy [5]. Model-based approaches use the generated residuals between the model outputs and actual measurements to detect and remove faults. For instance, in [14], the Nadaraya–Watson estimator and a priori observations are used to validate sensor measurements. Similarly, a priori system model information as a reference is used to detect failures in filtered state estimates [15–17]. However, requirement of the prior information restricts the usage of these methods in the general case where prior information is not available or unmodeled failure occurs. A method to detect spurious data based on the Bayesian framework is proposed in [18,19]. The method adds a term to the Bayesian formulation which has the effect of increasing the covariance of the fused probability distribution when measurement from one of the sensor is inconsistent with respect to the other. However, the method is based on heuristics and assumes independence of sensor estimates in its analysis. In [20], the Covariance Union (CU) method is proposed where the fused covariance is enlarged to cover all local means and covariances in such a way that the fused estimate is consistent under spurious data. However, the method incurs high computational cost and results in an inappropriately large conservative fused result.

In some applications, the state variables observed in a multisensory system may be subject to additional constraints. These constraints can arise due to the basic laws of physics, kinematics or geometry consideration of a system or due to the mathematical relations to satisfy among states. For instance, the energy conservation laws in an undamped mechanical system; Kirchhoff's laws in electric circuits; a road constraint in a vehicle-tracking scenario [21]; an orthonormal constraint in quaternion-based estimation [22] etc. These constraints if properly included can lead to improvement in state estimation and data fusion.

Various methods have been proposed to incorporate linear constraints among the state variables of dynamic systems [23–28]. For instance, the dimensionality reduction method converts a constrained estimation problem to an unconstrained one of lower dimension by eliminating some state variables using the constraints [25]. However, the state variables in a reduced dimension model may become difficult to interpret and their physical meaning may be lost [23]. The pseudo-measurement method satisfies the linear constraints among state variables by treating the state constraints as additional perfect/noise-free measurements [26–28]. However, this method increases the computational complexity of state estimation due to an increase in the dimension of augmented measurement. Furthermore, due to the singularity of augmented measurement covariance, the method may cause numerical problems [23,29]. A popular approach, the estimate projection method, projects the unconstrained estimate obtained from conventional Kalman filtering onto the constraint subspace using classical optimization methods [23,24]. Unfortunately, the method may not lead to the true constraint optimum since the projection method merely gives the solution as a feasible point that is closest to the unconstrained minimum.

This paper presents a unified and general data fusion framework, referred to as the Covariance Projection (CP) method to fuse multiple data sources under arbitrary correlations and linear constraints as well as data inconsistency. The method projects the probability distribution of true states and measurements around the predicted states and actual measurements onto the constraint manifold representing the constraints to be satisfied among true states and measurements. The proposed method also provides a framework for identifying and removing outliers in a fusion architecture where only sensor estimates may be available at the fusion center. This paper is an extended and improved version of the conference paper [30]. What was presented in [30] is a preliminary new framework of data fusion that we proposed. On the other hand, what is presented here represents a much more detailed implementation and refinement of the concept proposed in the conference paper. Specifically, this paper includes the following additions: (1) a detailed analysis of the equivalence of the proposed method to conventional methods for fusing redundant data sources; (2) handling linear constraints simultaneously under the proposed data fusion framework; (3) refining the mathematical formula and technical descriptions associated with them; and (4) detailed analysis of the method with additional simulations that deal with state estimation and data fusion in the presence of correlations, outliers and constraints.

## 2. Problem Statement

Consider a distributed sensor architecture [1], where each sensor system is equipped with a tracking system to provide local estimates of some quantity of interest in the form of mean and covariance. Assume the following linear dynamic system model for each local sensor system,

$$x_k = A_{k-1}\, x_{k-1} + B_{k-1} u_{k-1} + w_{k-1} \tag{1}$$

where $k$ is the discrete time, $A_k$ is the system matrix, $B_k$ is the input matrix, $u_k$ is the input vector and $x_k$ is the state vector. The system process is affected by zero mean Gaussian noise $w_k$ with covariance matrix $Q$. The sensor measurements are approximated as,

$$z_{k_i} = H_{k_i} x_k + v_{k_i}, i = 1, \dots, n \tag{2}$$

where $H_k$ is the observation matrix and $n$ represents the number of sensors. $v_{k_i}$ is Gaussian noise with covariance matrices $R_i, i = 1,2, \dots, n$. Each sensor systems employs a Kalman filter to provide local state mean estimate $\hat{x}_k$ and its covariance $P_k$ [31]. A prediction of the state estimate $\hat{x}_k^-$ and its estimation error covariance $P_k^-$ can be computed based on process model (1),

$$\hat{x}_k^- = A_{k-1}\, x_{k-1} + B_{k-1} u_{k-1} \tag{3}$$

$$P_k^- = A_{k-1} P_{k-1} A_{k-1}^T + Q_{k-1} \tag{4}$$

The Kalman filter then provides the state estimate $\hat{x}_k$ and its covariance $P_k$ as,

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H_k \hat{x}_k^-) \tag{5}$$

$$P_k = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T \tag{6}$$

with the Kalman gain, $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$. The estimates provided by sensor systems are assumed to be correlated due to the common process noise or double counting, that is, $P_{ij} = cov(\hat{x}_i, \hat{x}_j) \neq 0$. To ensure optimality of the fused results, the cross-covariance $P_{ij}$ should be properly incorporated.

Due to the inherent nature of the sensor and environmental factors [32], the sensor measurements can also be perturbed by unmodel random faults $e_{k_i}$,

$$z_{k_i} = H_i x_k + v_{k_i} + e_{k_i}, i = 1, \dots, n \tag{7}$$

Subsequently, the estimates computed by local sensor systems may be spurious and inconsistent. Therefore, validation of the sensor estimates is required to remove inconsistencies before the fusion process.

In addition, the states $x_k$ of the sensor systems may subject to linear constraints due to the geometry of the system environment or the mathematical description of the system [23], such that,

$$Cx_k = c \tag{8}$$

where $C \in \mathbb{R}^{n \times m}$ and $c \in \mathbb{R}^n$ are both known. $C$ is assumed to have a full row rank. The constraints provide deterministic information about the state variables and can be used to improve the fusion accuracy.

These issues of correlations, data inconsistency and state constraints motivate the development of the Covariance Projection method, which is described next.

## 3. Proposed Approach

The proposed method first represents the probability of true states and measurements in the extended space around the data from state predictions and sensor measurements, where the extended space is formed by taking states and measurements as independent variables. Any constraints among true states and measurements that should be satisfied are then represented as a constraint manifold in the extended space. This is shown schematically in Figure 1a for filtering as an example (refer to Equations (1)–(6)). Data fusion is accomplished by projecting the probability distribution of true states and measurements onto the constraint manifold.

More specifically, consider two mean estimates, $\hat{x}_1$ and $\hat{x}_2$, of the state $x \in \mathbb{R}^N$, with their respective covariances as $P_1$, $P_2 \in \mathbb{R}^{N \times N}$. Furthermore, the estimates are assumed to be correlated with cross-covariance $P_{12}$. The mean estimates and their covariances together with their cross-covariance in $\mathbb{R}^N$ are then transformed to an extended space of $\mathbb{R}^{2N}$ along with the linear constraint between the two estimates:
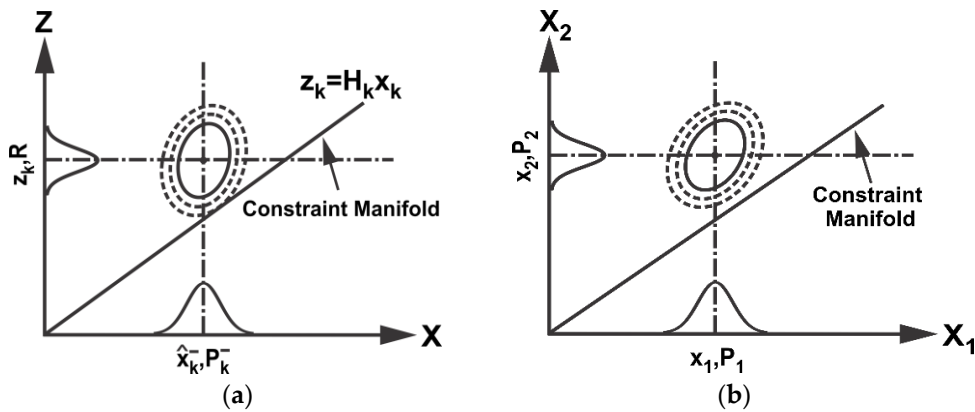


**Figure 1.** (**a**) Probability of true states and measurements in the extended space around the data from state predictions and sensor measurements and constraint manifold (**b**) Extended space representation of two data sources with constraint manifold.

$$\hat{x} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}, \quad P = \begin{bmatrix} P_1 & P_{12} \\ P_{12}^T & P_2 \end{bmatrix}, \quad C_1 \hat{x}_1 = C_2 \hat{x}_2 \tag{9}$$

where $C_1$ and $C_2$ are constant matrices of compatible dimensions. In the case where $\hat{x}_1$ and $\hat{x}_2$ estimate the same entity, $C_1$ and $C_2$ become identity matrix $I$. Figure 1b illustrates schematically the fusion of $\hat{x}_1$ and $\hat{x}_2$ in the extended space based on the proposed CP method. Fusion takes place by finding the point on the constraint manifold that represents the minimum weighted distance from $\hat{x}$ in $\mathbb{R}^{2N}$, where the weight is given by $P$. As seen later, the proposed CP method with the minimum weighted distance is shown to be equivalent to the minimum variance estimates but advantageous for dealing with additional constraints and data inconsistency.

To find a point on the constraint manifold with minimum weighted distance, we apply the whitening transform (WT) defined as, $W = D^{-1/2}E^T$, where $D$ and $E$ are the eigenvalue and eigenvector matrices of $P$. Applying WT,

$$\hat{x}^W = W\hat{x}, \ P^W = WPW^T, \ M^W = WM$$

where the matrix $M = [C_1 \ C_2]^T$ is the subspace of the constraint manifold. Figure 2 illustrates the transformation of the probability distribution as an ellipsoid into a unit circle after WT. The probability distribution is then orthogonally projected on the transformed manifold $M^W$ to satisfy the constraints between the data sources in the transformed space as illustrated in Figure 2. Inverse WT is applied to obtain the fused mean estimate and covariance in the original space,

$$\tilde{x} = W^{-1}P_r W\hat{x} \tag{10}$$

$$\tilde{P} = W^{-1}P_r P_r{}^T W^{-T} \tag{11}$$

where $P_r = M^W\left(M^{W^T}M^W\right)^{-1}M^{W^T}$ is the orthogonal projection matrix. Using the definition of various components in (10) and (11), a close form simplification can be obtained as,

$$\tilde{x} = M(M^T P^{-1}M)^{-1}M^T P^{-1}\hat{x} \tag{12}$$

$$\tilde{P} = M(M^T P^{-1}M)^{-1}M^T \tag{13}$$

The details of the simplification are provided in Appendix A. Due to the projection in extended space of $\mathbb{R}^{2N}$, (12) and (13) provide a fused result with respect to each data source. In the case where $\hat{x}_1$ and $\hat{x}_2$ estimate the same entity, that is, $M = [I_N \ I_N]^T$, the fused result will be same for the two data sources. As such, a close form equation for fusing redundant data sources in $\mathbb{R}^N$ can be obtained from (12) and (13) as,

$$\tilde{x} = (M^T P^{-1}M)^{-1}M^T P^{-1}\hat{x} \tag{14}$$
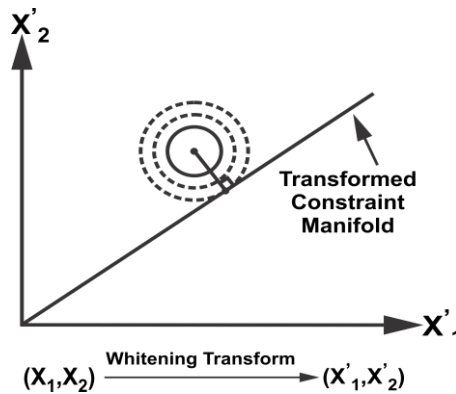
$$\tilde{P} = (M^T P^{-1}M)^{-1} \tag{15}$$



**Figure 2.** Whitening transform and projection.

Given $n$ mean estimates $\hat{x}_1$, $\hat{x}_2$, ..., $\hat{x}_n$ of a state $x \in \mathbb{R}^N$ with their respective covariances $P_1, P_2, \dots, P_n \in \mathbb{R}^{N \times N}$ and known cross-covariances $P_{ij}, i, j = 1, \dots, n$, (14) and (15) can be used to obtain the optimal fused mean estimate and covariance with $M = [I_{N1}, I_{N2}, \dots, I_{Nn}]^T$.

For fusing correlated estimates from $n$ redundant sources, the CP method is equivalent to the weighted fusion algorithms [9,10], which compute the fused mean estimate and covariance as a summation of weighted individual estimates as,

$$\tilde{x} = \sum_{i=1}^{n} c_i \hat{x}_i \tag{16}$$

$$\tilde{P} = \sum_{i,j=1}^{n} c_i P_{ij} c_j^T \tag{17}$$

with $\sum_{i=1}^{n} c_i = I$. Where the weights $c_i$ are determined by solving some cost function of (16) and (17) such that, $\sum_{i=1}^{n} c_i = I$. Equivalently, the CP fused mean and covariance can be written as,

$$\tilde{x} = L\hat{x} \tag{18}$$

$$\tilde{P} = LPL^T \tag{19}$$

where $L = [L_1, L_2, \dots, L_n] = (M^T P^{-1} M)^{-1} M^T P^{-1}$ and $\sum_{i=1}^{n} L_i = I$. In the particular case of two data sources, the CP fused solution reduces to the well-known Bar-Shalom Campo formula [7],

$$\tilde{x} = (P_2 - P_{21})(P_1 + P_2 - P_{12} - P_{21})^{-1}\hat{x}_1 + (P_1 - P_{12})(P_1 + P_2 - P_{12} - P_{21})^{-1}\hat{x}_2 \tag{20}$$

$$\tilde{P} = P_1 - (P_1 - P_{12})(P_1 + P_2 - P_{12} - P_{21})^{-1}(P_1 - P_{21}) \tag{21}$$

Although equivalent to the traditional approaches in fusing redundant data sources, the proposed method offers a generalized framework not only for fusing correlated data sources but also for handling linear constraints and data inconsistency simultaneously within the framework.

The proposed method provides an unbiased and optimal fused estimate in the sense of minimum mean square error (MMSE).

**Theorem 1.** *For $n$ unbiased mean estimates $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n$, the fused mean estimate $\tilde{x}$ provided by the CP method is an unbiased estimator of $x$, that is, $E(\tilde{x}) = E(x)$.*

**Proof.** From (18), we can write,

$$\tilde{x} = [L_1, L_2, \dots, L_n] \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \vdots \\ \hat{x}_n \end{bmatrix}$$

$$\tilde{x} = L_1 \hat{x}_1 + L_2 \hat{x}_2 + \dots + L_n \hat{x}_n$$

$$\tilde{x} = \sum_{i=1}^{n} L_i \hat{x}_i$$

Taking the expectation on both sides, we get,

$$E(\tilde{x}) = E\left(\sum_{i=1}^{n} L_i \hat{x}_i\right)$$

$$E(\tilde{x}) = \sum_{i=1}^{n} L_i E(\hat{x}_i)$$

Since the sensor estimates $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n$ are unbiased, we have $E(\hat{x}_1) = E(\hat{x}_2) = \dots E(\hat{x}_n) = \mathrm{E}(x)$,

$$E(\tilde{x}) = E(x)$$

where $\sum_{i=1}^{n} L_i = I$. This concludes that the fused state estimate $\tilde{x}$ is an unbiased estimate of $x$. □

**Theorem 2.** *The fused covariance $\tilde{P}$ of the CP method is smaller than the individual covariances, that is, $\tilde{P} \leq P_i, i = 1, 2, \dots, n$.*

**Proof.** From equation (15), we can write,

$$\tilde{P} = (M^T P^{-1} M)^{-1}$$

By Schwartz matrix inequality, we have,

$$\tilde{P} = [(P^{-\frac{1}{2}}M)^T (P^{\frac{1}{2}}M_i)]^T \times [(P^{-\frac{1}{2}}M)^T (P^{-\frac{1}{2}}M)]^{-1} \times [(P^{-\frac{1}{2}}M)^T (P^{\frac{1}{2}}M_i)]$$

$$\leq \left(P^{\frac{1}{2}}M_i\right)^T \left(P^{\frac{1}{2}}M_i\right) = P_i$$

where $M$ is the constraint between data sources and $M_i = [I_{Ni}, 0, \dots, 0]^T$ is the constraint matrix for $P_i$. The equality holds for $P_i = P_{ij}$, that is, $\tilde{P} = P_i$, when $P_i = P_{ij}, j = 1, 2, \dots, n$.

It can be observed from (14) and (15) that computation of cross-covariance $P_{ij}$ is needed to compute the fused mean and covariance. Cross-covariance among the local estimates can be computed as [9,10,33],

$$P_{ij} = [I - K_i H_i][AP_{ij}^{k-1}A^T + BQB^T][I - K_j H_j]^T \tag{22}$$

where $K_i$ and $K_j$ are the Kalman gain of source $i$ and $j$ respectively for $i, j = 1, \dots, n$ and $P_{ij}^{k-1}$ represent the cross covariance of the previous cycle between source $i$ and $j$. □

## 4. Fusion in the Presence of Spurious Data

Due to the inherent nature of sensor devices and the real-world environment, the sensor observations may also be affected by random faults. Subsequently, the local estimates provided by sensor systems in a distributed architecture may be spurious and inconsistent. This may cause the fusion methodologies to fail since they are based on the assumption of consistent input sensor estimates. Therefore, a validation scheme is required to detect and remove the spurious estimates from the fusion pool. The proposed approach exploits the constraint manifold among sensor estimates to identify any data inconsistency. The identification of inconsistent data is based on the distance from the constraint manifold to the mean of redundant data sources in the extended space that provides a confidence measure with the relative disparity among data sources. Assuming a joint multivariate normal distribution for the data sources, the data confidence can be measured by computing the distance from the constraint manifold as illustrated in Figure 3.
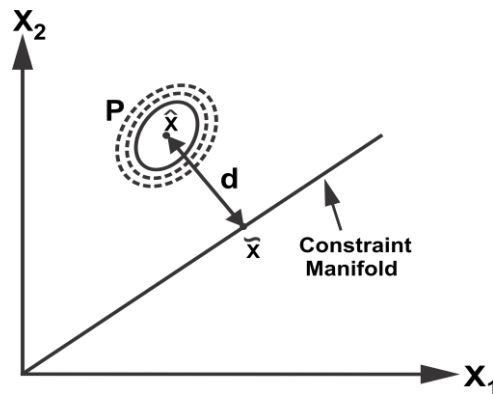


**Figure 3.** The distance of the multi-variate distribution from the constraint manifold.

Consider the joint space representation of $n$ sensor estimates $(\hat{x}_{N1}, P_1), (\hat{x}_{N2}, P_n), \dots, (\hat{x}_{Nn}, P_1)$,

$$\hat{x} = \begin{bmatrix} \hat{x}_{N1} \\ \hat{x}_{N2} \\ \vdots \\ \hat{x}_{Nn} \end{bmatrix}, \quad P = \begin{bmatrix} P_1 & P_{12} & \cdots & P_{1n} \\ P_{12}^T & P_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ P_{1n}^T & \cdots & \cdots & P_{nn} \end{bmatrix}$$

where $N$ is the dimension of the state vector. The distance $d$ from the constraint manifold can be calculated as,

$$d = (\hat{x} - \tilde{x})^T P^{-1} (\hat{x} - \tilde{x}) \tag{23}$$

where $\tilde{x}$ is the point on the manifold and can be obtain by using (12). In the case of two data sources with mean $\hat{x}_1, \hat{x}_2 \in \mathbb{R}^N$ and respective covariance matrices $P_1$ and $P_2 \in \mathbb{R}^{N \times N}$. The distance $d$ can be obtained as,

$$d = [(\hat{x}_1 - \tilde{x})^T \quad (\hat{x}_2 - \tilde{x})^T] \begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix}^{-1} \begin{bmatrix} \hat{x}_1 - \tilde{x} \\ \hat{x}_2 - \tilde{x} \end{bmatrix}$$

The point on the manifold is given as,

$$\tilde{x} = P_2(P_1 + P_2)^{-1}\hat{x}_1 + P_1(P_1 + P_2)^{-1}\hat{x}_2$$

Simplifying, we get,

$$d = [\hat{x}_1 - \hat{x}_2]^T (P_1 + P_2)^{-1} [\hat{x}_1 - \hat{x}_2] \tag{24}$$

The details of simplifications are provided in Appendix B. From (24), it can be observed that distance $d$ is a weighted distance between the two data sources and it can provide a measure of nearness or farness of the two data sources to each other. A large value of $d$ implies a large separation while a small $d$ signifies closeness of the data sources. In other words, the distance from the manifold provides an indication of the relative disparity among data sources.

**Theorem 3.** *For $n$ data sources of $N$ dimension, the $d$ distance (23) follow a chi-squared distribution with $nN$ degrees of freedom (DOF), that is, $d \sim \chi^2(Nn)$.*

**Proof:** From (23) we have,

$$d = (\hat{x} - \tilde{x})^T P^{-1} (\hat{x} - \tilde{x}) \tag{25}$$

Under the whitening transformation, $WP^{-1}W = I$, $W\hat{x} = \hat{x}^W$ and $W\tilde{x} = \tilde{x}^W$. Thus, we can write,

$$(\hat{x} - \tilde{x})^T P^{-1} (\hat{x} - \tilde{x}) = (\hat{x}^W - \tilde{x}^W)^T (\hat{x}^W - \tilde{x}^W)$$
$$\Rightarrow (W(\hat{x} - \tilde{x}))^T (W(\hat{x} - \tilde{x})) = y^T y \tag{26}$$

where for normal distribution $(\hat{x} - \tilde{x})$, $y = W(\hat{x} - \tilde{x})$ is an independent standard normal distribution $\mathcal{N}(0,1)$. For $N$ dimensions of the state vector, the right-hand side of (26) is $\sum_{i=1}^{N} y_i^2$, thus distance d follows a chi-square distribution with $N$ DOF, that is, $d \sim \chi^2(N)$. For $n$ data sources with $N$ states,

$$d \sim \chi^2(Nn)$$

Since $d$ is a chi-square distribution with $Nn$ DOF, then for any significance level $\alpha \in (0,1)$, $\chi_\alpha^2(Nn)$ is defined such that the probability,

$$P\{d \geq \chi_\alpha^2(Nn)\} = \alpha$$

This is depicted in Figure 4. Hence, to have a confidence of $100 \times (1 - \alpha)$ percent, $d$ should be less than respective critical value as illustrated in Figure 4. A chi-square table [34] can be used to obtain the critical value for the confidence distance with a particular significance level and DOF. A value of $\alpha = 0.05$ is assumed in this paper unless specified. □
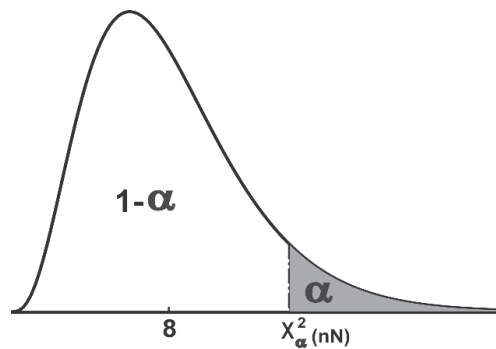


**Figure 4.** Chi-square distribution with 8 degreed of freedom. The unshaded area represents a cumulative probability associated with chi-square statistics $\chi_\alpha^2(Nn)$.

*4.1. Inconsistency Detection and Exclusion*

To obtain reliable and consistent fusion results, it is important that the inconsistent estimates be identified and excluded before fusion. For this reason, at each time step when the fusion center receives computed estimates from sensor nodes, distance $d$ is calculated. A chi-square table is then used to obtain the critical value for a particular significance level and DOF. A computed distance $d$ less than the critical value mean that we are confident about the closeness of sensor estimates and

that they can be fused together to provide a better estimate of the underlying states. On the other hand, a distance $d$ greater than or equal to the critical value indicate spuriousness of the sensor estimates. At least one of the sensor estimate is significantly different than the other sensor estimates. To exclude the outliers, a distance from the manifold is computed for every estimate and compared with the respective critical values. For $n$ mean estimates $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n$ with respective covariances $P_1, P_2, \dots, P_n$ and cross-covariances $P_{ij}$ for $i, j = 1, \dots, n$, the hypothesis and decision rule are summarized as follows,

**Hypotheses:**

$$H_0: \hat{x}_1 = \hat{x}_2 = \cdots = \hat{x}_n$$

$$H_1: \hat{x}_1 \neq \hat{x}_2 \neq \cdots \neq \hat{x}_n$$

**Compute:** $d = (\hat{x} - \tilde{x})^T P^{-1} (\hat{x} - \tilde{x})$

**Decision Rule:** Accept $H_0$ if $d < \chi_\alpha^2(Nn)$

Reject $H_0$ if $d \geq \chi_\alpha^2(Nn)$

If the hypothesis $H_0$ is accepted, then the estimates are optimally fused using (14) and (15). On the other hand, rejection of null hypothesis means that at least one of the sensor estimates is significantly different than the other estimates. Then, a distance from the manifold is computed for each of the estimates as,

$$d_i = (\hat{x}_i - \tilde{x})^T P_i^{-1} (\hat{x}_i - \tilde{x}), i = 1,2, \dots, n$$

with $\tilde{x}$ computed using (14). The outliers are identified and eliminated based on the respective critical value, that is, if $d_i \geq \chi_\alpha^2(N)$, they are rejected, where $N$ is the dimension of an individual data source.

*4.2. Effect of Correlation on d Distance*

Since the estimates provided by multiple data sources are correlated, it is important to consider the effect of cross-correlation in the calculation of confidence distance. Consider two sensor estimates $\hat{x}_1 \in \mathbb{R}^1$ and $\hat{x}_2 \in \mathbb{R}^1$ with respective variances $\sigma_1^2$ and $\sigma_2^2$ and cross-covariance $\sigma_{12}^2 = \rho \sqrt{\sigma_1^2 \sigma_2^2}$, where $\rho \in [-1,1]$ is the correlation coefficient. The $d$ distance for the pair of multivariate Gaussian estimates $(\hat{x}_1, \sigma_1^2)$ and $(\hat{x}_2, \sigma_2^2)$, with cross-covariance $\sigma_{12}^2$ can be written as,

$$d = \frac{[\hat{x}_1 - \hat{x}_2]^2}{\sigma_1^2 + \sigma_2^2 - \sigma_{12}^2 - \sigma_{21}^2} \tag{27}$$

It is apparent that the distance between the mean values is affected by the correlation between the data sources. Figure 5 illustrates the dependency of confidence distance $d$ on the correlation coefficient. Figure 5a shows the distance $d$ with changing correlation coefficient from −1 to 1. It can be observed that a positive correlation between the data sources results in a large $d$ distance. This means that a slight separation between the positively correlated data sources indicates spuriousness with high significance as compared to negatively correlated and uncorrelated data sources. Figure 5b shows the scenario in which a data source (with changing mean and constant variance) is moving away from another data source (with constant mean and constant variance). The distance $d$ is plotted for various values of correlation coefficients. The y-axis shows the percentage of rejection of the null hypothesis $H_0$. It can be noted that ignoring the cross-correlation in distance $d$ results in underestimated or overestimated confidence and may lead to an incorrect rejection of the true null hypothesis (Type I error) or incorrect retaining of false null hypothesis (Type II error). The proposed framework inherently takes care of any cross-correlation among multiple data sources in the computation of distance $d$.
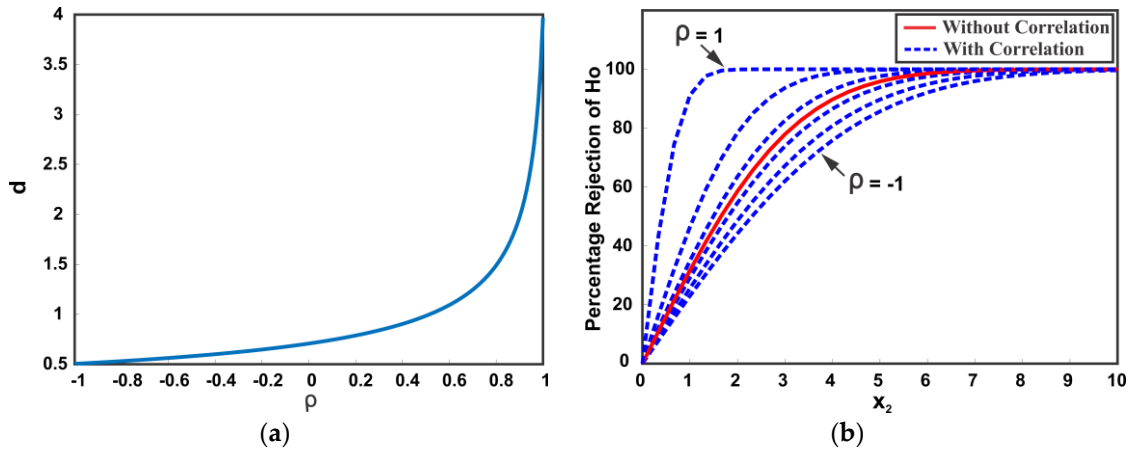
**Figure 5.** Effect of correlation on $d$ distance (**a**) $d$ distance with correlation $\rho \in [-1,1]$; (**b**) percentage of rejecting the null hypothesis $H_0$ with different correlation values.

**Example:** *Consider a numerical simulation with the constant state,*

$$x_k = 10$$

Three sensors are used to estimate the state $x_k$, where the measurements of the sensors are affected with respective variances of $R_1$, $R_2$ and $R_3$. The values for the parameters assumed in the simulation are,

$$Q = 2, R_1 = 0.5, R_2 = 1, R_3 = 0.9$$

The sensor measurements are assumed to be cross-correlated. It is also assumed that the sensor 1, sensor 2 and sensor 3 measurements are independently corrupted by unmodeled noise and produce inconsistent data for 33%, 33% and 34% of the time respectively. The sensors compute local estimates of the state and send it to the fusion center. Three strategies for combining the local sensor estimates are compared: (1) CP, which fuses the three sensor estimates using (14) and (15) without removing outliers; (2) CP WO-d means the outliers were identified and rejected before fusion based on (27) with $\sigma_{12}^2 = 0$, that is, correlation in computation of $d$ is ignored and; (3) CP WO-dC, reject the outliers based on (27) with taking into account the cross-correlation. Figure 6 shows the fused solution of three sensors when the estimate provided by sensor 2 is in disagreement with sensor 1 and 3. It can be observed from Figure 6 that neglecting the cross-correlation in CP WO-d results in Type II error, that is, all the three estimates are fused despite the fact that estimate 2 is inconsistent. CP WO-dC correctly identifies and eliminates the spurious estimate before the fusion process. Figure 7 shows the estimated position after the fusion of the three sensors' estimates for 100 samples. It can be seen that the presence of outliers greatly affects the outcome of multisensor data fusion. As depicted in Figure 7, eliminating outliers before fusion improves the estimation performance. The fused samples of CP WO-d and CP WO-dC on average lies closer to the actual state. Figure 7 also shows the difference in fusion performance when outliers are identified with and without considering cross-correlation. It can be noted that neglecting the correlation affects the estimation quality because of Type I and Type II errors.
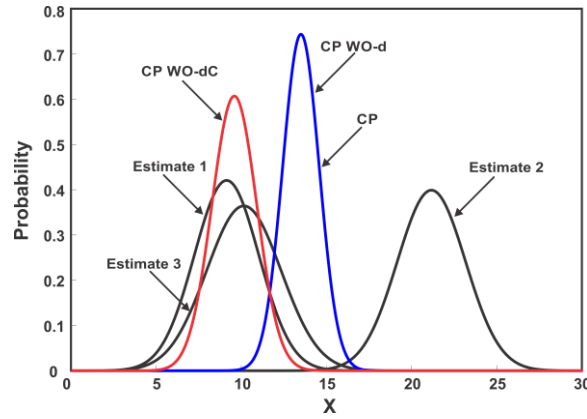
**Figure 6.** Three-sensor fusion when the estimate of sensor 2 is inconsistent. Neglecting the cross-correlation results in Type II error.
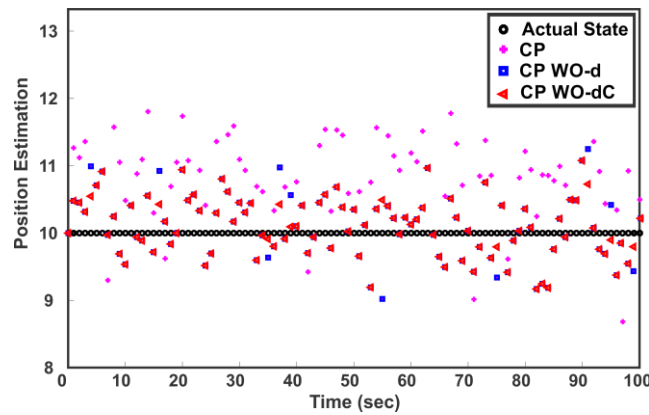


**Figure 7.** Estimated position after three-sensor fusion in presence of inconsistent estimates.

## 5. Fusion under Linear Constraints

The system model of a linear dynamic system takes into account the relation and dependencies among components of the state vector. In some applications, however, the state variables may be subject to additional constraints due to the basic laws of physics, geometry of the system environment or due to the mathematical description of the state vector. Imposing such certain information in an otherwise probabilistic setting should yield a more accurate estimate that is guaranteed to be feasible.

Consider a linear dynamic system model,

$$x_k = A_{k-1} x_{k-1} + B_{k-1} u_{k-1} + w_{k-1} \tag{28}$$

$$z_{k_i} = H_{k_i} x_k + v_{k_i}, i = 1, \dots, n \tag{29}$$

where $k$ represents the discrete-time index, $A_k$ is the system matrix, $B_k$ is the input matrix, $u_k$ is the input vector and $x_k$ is the state vector. The system process noise $w_k$ with covariance matrix $Q$ and measurement noise $v_k$ with covariance $R$ are assumed to be correlated, that is,

$$E\left\{ \begin{bmatrix} w_k \\ v_k \end{bmatrix} \begin{bmatrix} w_k \\ v_k \end{bmatrix}^T \right\} = \begin{bmatrix} Q & P_{QR} \\ P_{QR}^T & R \end{bmatrix}$$

The state $x_k \in \mathbb{R}^N$ is known to be constrained as,

$$C x_k = c = 0 \tag{30}$$

For $c \neq 0$, the state space can be translated by a factor $c$ such that $C\bar{x}_k = 0$. After constrained state estimation, the state space can be translated back by the factor c to satisfy $C x_k = c$. Hence, without loss of generality, the $c = 0$ case is considered for analysis here. The matrix $C \in \mathbb{R}^{n \times m}$ is assumed to have a full row rank. A row deficient matrix $C$ signifies the presence of redundant constraints. In such a case, we can simply remove the linearly dependent rows from $C$. In the

following, the estimate projection (EP) method is briefly reviewed which is followed by the Covariance Projection (CP) method for linear constraints among state variables.

### 5.1. Estimate Projection Method

The estimate projection (EP) method [23,24] projects the unconstrained estimate obtained from Kalman filtering onto the constraint subspace to satisfy the linear constraints among state variables. Let us denote the unconstrained filtered estimate and constrained estimate as $(\hat{x}^u, P^u)$ and $(\hat{x}^p, P^p)$, respectively. Then the following optimization problem is solved to obtain the constrained estimate,

$$\min_{\hat{x}^p}(\hat{x}^p - \hat{x}^u)^T U (\hat{x}^p - \hat{x}^u) \text{ such that } C\hat{x}^p = 0 \tag{31}$$

where $U$ is any symmetric positive definite weighting matrix. Solving (31) using Lagrange multipliers results in a constrained mean and covariance,

$$\hat{x}^p = J\hat{x}^u \tag{32}$$

$$P^p = JP^uJ^T \tag{33}$$

where $J$ is the projector on the null space of constrained matrix $C$, defined as,

$$J = I - U^{-1}C^T(CU^{-1}C^T)^{-1}C$$

Any symmetrical positive definite matrix can be used as a weighting matrix $U$ to obtain the constrained estimate but the two most common choices are identity matrix $I$ and inverse of unconstrained covariance $P^{u-1}$.

### 5.2. Covariance Projection Method for Linear Constraints

The CP framework incorporates any linear constraints among states without any additional processing. Let us denote the constrained filtered estimate of the CP method as $(\hat{x}^c, P^c)$. The extended space representation of the state predictions and measurements of multiple sensors can be written as,

$$\hat{x} = \begin{bmatrix} \hat{x}_k^- \\ z_{k_1} \\ \vdots \\ z_{k_n} \end{bmatrix}, \; P = \begin{bmatrix} P_k^- & P_{P_k^- R_1} & \cdots & P_{P_k^- R_n} \\ P_{P_k^- R_1}^T & R_1 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ P_{P_k^- R_n}^T & \cdots & \cdots & R_n \end{bmatrix}$$

Then the CP estimate in the presence of linear constraints among states can be obtained using (12) and (13) as,

$$\hat{x}^c = M_c(M_c^T P^{-1} M_c)^{-1} M_c^T P^{-1} \hat{x} \tag{34}$$

$$P^c = M_c(M_c^T P^{-1} M_c)^{-1} M_c^T \tag{35}$$

where the $M_c$ matrix is the subspace of the constraint among the state prediction $x_k^-$ and sensor measurements $z_{k_i}$ as well as linear constraints $C$ among state variables. The subspace of the linear constraint among state prediction and sensor measurements can be written as,

$$M = [I_N, H_1, H_2, \dots, H_n]^T$$

Then, $M_c$ is a combination of $M$ and $C$, that is,

$$M_c \in (M, C)$$

The projection of the probability distribution of true states and measurements around the predicted states and actual measurements onto the constraint manifold $M_c$ in the extended space provide the filtered or fused estimate of state prediction and sensor measurements as well as completely satisfying the linear constraints among the states directly in one step.

It can be observed from (31) that the EP method forces the unconstrained estimate on the linear constraint under some norm to satisfy the linear constraints among state variables. Consequently, the true optimality cannot be guaranteed due to the fact that the projected point close to the

unconstrained estimate does not imply that it is close to the true constrained state [29]. On the other hand, the unified constraint matrix $M_c$ of the CP method ensures that the linear constraints among state variables are exactly satisfied. Furthermore, as compared to the EP method that needs the online projection steps (32) and (33) after filtering, the $M_c$ matrix for the CP method can be computed offline. This means that the EP method is computationally less efficient. Additionally, the proposed CP method is inherently suitable for taking care of any cross-correlation in the constrained estimation process.

## 6. Simulation Results

In this section, illustrative examples are provided to demonstrate the effectiveness of the theoretical results derived in the previous sections. We use the Monte Carlo technique [35], a method extensively used in a wide variety of fields such as physical science, computational biology, statistics, computational geometry, artificial intelligence, engineering, decision theory, and quantitative finance (see, e.g., the recent works [36–40]).

### 6.1. Tracking in the Presence of Correlations and Outliers

Consider a target tracking scenario characterized by the following dynamic system model,

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} w_{k-1} \tag{36}$$

with the state vector $x_k = [x, \dot{x}]^T$. Where $x$ and $\dot{x}$ are the position and velocity of the target at time $k$, respectively. $T = 0.5$ s is the sampling period. The system process is affected by zero mean Gaussian noise with covariance matrix $Q$. Four sensors are employed to track the movement of the target, where the sensor measurements are approximated by the following equation,

$$z_{k_i} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k + v_{k_i}, i = 1,2,3,4 \tag{37}$$

The measurements of the sensors are affected by noise $v_{k_i}$ with respective covariances of $R_1$, $R_2$, $R_3$ and $R_4$. The covariances of the process noise and sensor measurement noises used in the simulation are,

$$Q = 3.5, R_1 = diag(5,3.5), R_2 = diag(2,8),$$

$$R_3 = diag(7,2.1), \ R_4 = diag(2.5,5)$$

Starting from an initial value of [100, 3], in each time step the individual sensor uses (36) to predict the state of the target and then update the state prediction by its own sensor measurements. The local estimates are assumed to be correlated and (22) is used to compute the cross-correlation among local estimates. The estimated mean and covariances of the states by each sensor are sent to the fusion center, where they are fused by the CP method. Table 1 summarizes the trace of covariance of local sensors along with the trace of fused covariance provided by the CP method. As seen from Table 1, the trace of covariance provided by the CP method is less than the individual local state estimates, that is, $trace\widetilde{P} \leq traceP_i$, $i = 1, \dots, 4$. This means that the fused result is better than each of the local state estimates.

In order to further verify this theoretical result, we compute the mean square error (MSE) as,

$$S_{MSE}(k) = \frac{1}{V} \sum_{i=1}^{V} [\hat{x}_i(k) - x_i(k)]^T [\hat{x}_i(k) - x_i(k)]$$

where $V$ is the number of Monte Carlo trials and $\hat{x}_i(k)$ and $x_i(k)$ are the estimated and actual state vector respectively. Since,

$$trace(P_i) = E[(\hat{x}_i - x)^T (\hat{x}_i - x)]$$

Then we have [41],

$$S_{MSE}(k) = trace(P_i), k \to \infty, V \to \infty \tag{38}$$

The simulation is carried out for 1000 Monte Carlo runs and the local estimates provided by four sensor nodes along with the fused result of the CP method are shown in Figure 8. The straight lines in Figure 8 denote the trace of error covariance matrices and the solid curve represents the MSE of local and fused estimates. It can be observed from Figure 8 that the MSE of the individual sensor node fluctuates around the trace, which is consistent with (38). Furthermore, the accuracy relation of local sensor estimates and fused estimates in terms of MSE in Figure 8 is coincident with the theoretical results in Table 1.
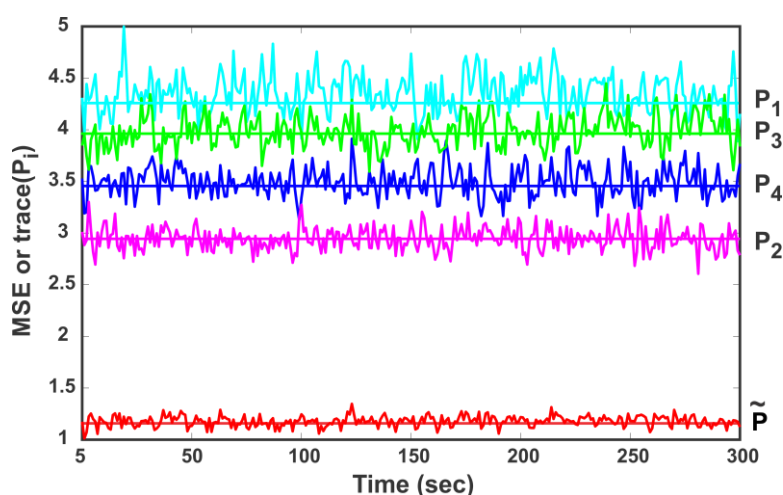


**Figure 8.** The mean square error (MSE) and trace ($P_i$) of local and fused estimates.

**Table 1.** The accuracy comparison in terms of trace of matrices.

| $traceP_1$ | $traceP_2$ | $traceP_3$ | $traceP_4$ | $trace\widetilde{P}$ |
|---|---|---|---|---|
| 4.3271 | 2.9109 | 3.9656 | 3.4321 | 1.2674 |

Consider the same dynamic example of four sensors with the following system and measurement equations,

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_{k-1} + w_{k-1} \tag{39}$$

$$z_{k_i} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k + v_{k_i} + e_{k_i}, i = 1,2,3,4 \tag{40}$$

where the process and measurement noise parameters are the same. Now, it is also assumed that the sensor 1, sensor 2, sensor 3 and sensor 4 measurements are independently affected by unmodeled random noise $e_{k_i}$ for 5%, 15%, 20% and 10% of the time, respectively, and thus the estimates provided by the sensors are spurious. The control input alternate between 1 and −1 and set to a value of 1 if $\dot{x} < 30$ otherwise it is changed to −1 until $\dot{x} < 5$. Starting from an initial value of [100, 3], in each time step the individual sensor node compute local filtered estimates. The estimated mean and covariances by each sensor are sent to the fusion center, where they are fused. The three fusion strategies of CP (fusion without outlier removal), CP WO-d (outlier removal without considering cross-correlation) and CP WO-dC (taking care of correlation in outlier removal) are compared based on root mean squared error (RMSE) between the actual state value and fused estimate of the state. The inconsistency is detected with significance level $\alpha$ = 0.05. Figure 9a,b illustrate the RMSE of the target position and velocity respectively versus time for 1000 Monte Carlo runs. Table 2 summarizes the average RMSE of position and velocity. It can be observed from Figure 9 and Table 2 that the presence of outliers deteriorates the performance of multisensor data fusion. Eliminating the outliers before fusion greatly improves the estimation quality. Figure 9 and Table 2 also shows the difference in fusion performance of CP WO-d and CP WO-dC, when outliers are identified with and without

consideration of cross-correlation in distance $d$ respectively. It can be noted that consideration of correlation in distance $d$ improves the estimation quality in presence of outliers by avoiding Type I and Type II errors.
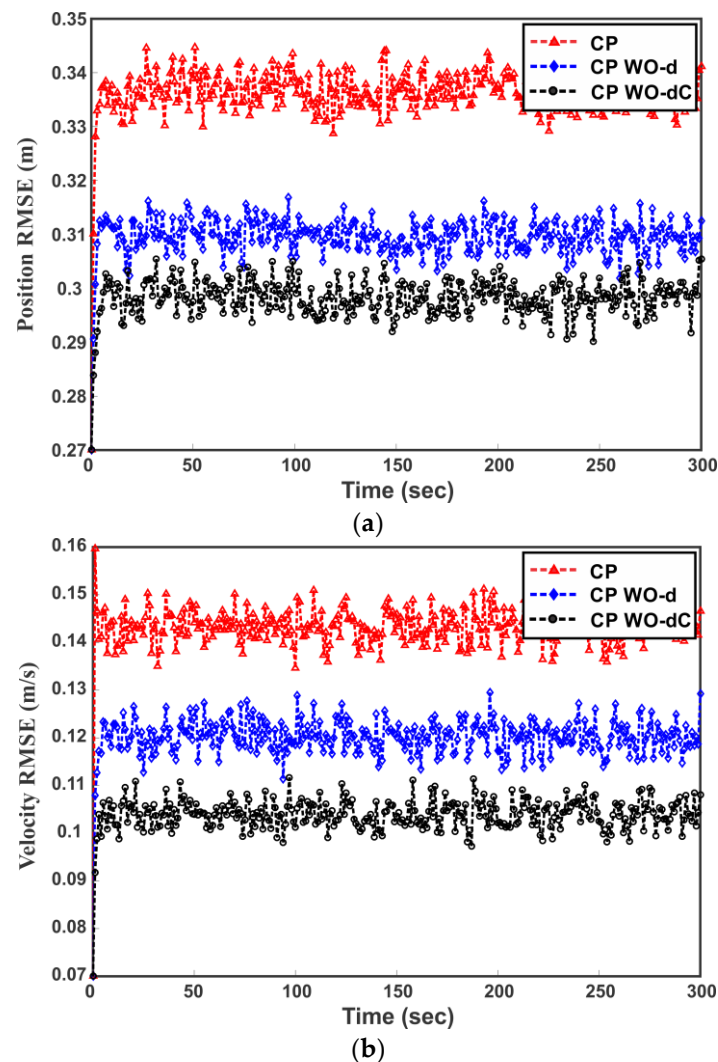


**Figure 9.** Illustration of multisensor data fusion in the presence of inconsistent estimates. (**a**) Position root mean squared error (RMSE); (**b**) velocity RMSE.

**Table 2.** Average RMSE for 1000 Monte Carlo Runs.

| Average RMSE | CP | CP WO-d | CP WO-dC |
|---|---|---|---|
| Position (m) | 101.1605 | 93.0824 | 89.6491 |
| Velocity (m/s) | 43.0339 | 36.0899 | 31.2033 |

*6.2. Target Tracking in the Presence of Linear Constraints*

Consider a 2D target tracking problem [24] with the following system equation,

$$x_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} 0 \\ 0 \\ Tsin\theta \\ Tcos\theta \end{bmatrix} u_{k-1} + w_{k-1} \tag{41}$$

where $x_k = [x_1, x_2, \dot{x}_1, \dot{x}_2]^T$ is the state vector, with the first two states as the north and east position with the last two states as the north and east velocity of the target respectively. A sensor measures the position and velocity of the system as,

$$z_k = Hx_k + v_k \tag{42}$$

with $H = \begin{bmatrix} 1 & 3 & 2 & 1 \\ 3 & 1 & 0 & 1 \\ 2 & 1 & 3 & 3 \\ 2 & 2 & 0 & 1 \end{bmatrix}$. $w_k$ and $v_k$ are the Gaussian process and measurement noise, respectively. Suppose that we have additional information that the vehicle is moving on the road with a heading angle of $\theta$ from the east $x_2$, then we can write,

$$\tan\theta = \frac{x_1}{x_2} \Rightarrow x_1 - x_2\tan\theta = 0$$

$$\tan\theta = \frac{\dot{x}_1}{\dot{x}_2} \Rightarrow \dot{x}_1 - \dot{x}_2\tan\theta = 0$$

Due to the heading of the vehicle, the states are dependent on each other, thus providing us additional constraints,

$$\begin{bmatrix} 1 & -\tan\theta & 0 & 0 \\ 0 & 0 & 1 & -\tan\theta \end{bmatrix} x_k = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{43}$$

Based on the constraint matrix $M = [I_4 \quad H]^T$ between the state prediction and sensor measurement and linear constraints (43) among the state variables, the unified constraint matrix $M_c$ can be obtained as,

$$M_c = \begin{bmatrix} \tan\theta & 1 & 0 & 0 & 3+\tan\theta & 1+3\tan\theta & 1+2\tan\theta & 2+2\tan\theta \\ 0 & 0 & \tan\theta & 1 & 1+2\tan\theta & 1 & 3+3\tan\theta & 1 \end{bmatrix}^T \tag{44}$$

The covariance of the process and measurement noise are assumed to be,

$$Q = \begin{bmatrix} 100 & 20 & 0 & 0 \\ 20 & 100 & 0 & 0 \\ 0 & 0 & 30 & 10 \\ 0 & 0 & 10 & 30 \end{bmatrix}, R = \begin{bmatrix} 500 & 10 & 20 & 30 \\ 10 & 500 & 15 & 10 \\ 20 & 15 & 100 & 20 \\ 30 & 10 & 20 & 100 \end{bmatrix}$$

The target is set to an initial state of $[0 \quad 0 \quad 15\tan\theta \quad 15]$. The sampling time $T$ is set to 1 s, heading angle $\theta = \pi/3$ and control input $u_k = 1$ if $\dot{x}_1$ or $\dot{x}_2 < 30$ otherwise it is changed to −1 until $\dot{x}_1$ or $\dot{x}_2 < 5$. The process and measurement noise are assumed to be correlated with covariance,

$$P_{QR} = \rho F_1 F_2^T$$

where $F_1$ and $F_2$ are the cholesky decomposition of $Q$ and $R$ respectively. $\rho$ is the correlation coefficient and assumed as 0.4 in the simulation.

Starting from the initial value, in each time step (41) is used to predict the states of the vehicle. The states are then updated with the sensor measurements (42). Using $M_c$ from (44) in (34) and (35), the proposed method directly satisfies the constraint between the perdition and measurement as well as the linear constraints among states due to the heading of the vehicle. On the other hand, the EP method first employs a Kalman filter to obtain the unconstrained state estimates and then project the unconstrained state estimates on the linear constraints subspace to satisfy the constraints among states. The performance of the proposed method is compared with the unconstrained estimate and estimate provided by the EP method in terms of RMSE. Table 3 summarizes the average RMSE of the individual states for 1000 Monte Carlo runs. It can be observed from Table 3 that the RMSE of the CP method is lower than the other estimators for all states. For the EP method, the use of $P^{u-1}$ as a weighting parameter provides better results than using $I$. The comparative results of different methods can be also seen in Figure 10a,b, where Figure 10a is the RMSE of the northerly position and Figure 10b is the RMSE of northerly velocity. It can be seen that the proposed method performs better as compared to the unconstrained state estimation and EP method. Figure 11a,b show the variance of northerly position and northerly velocity of the vehicle, respectively. It is proven that the EP method with weighting parameter $P^{u-1}$ has the smallest covariance [24]. This can also be observed from Figure 11a,b where the EP method with weight $P^{u-1}$ provides smaller variance than the unconstrained estimate and EP method by identifying matrix $I$ as the weighting parameter.

However, the CP method results in even smaller state variance than all the competing methods as seen in Figure 11a,b.
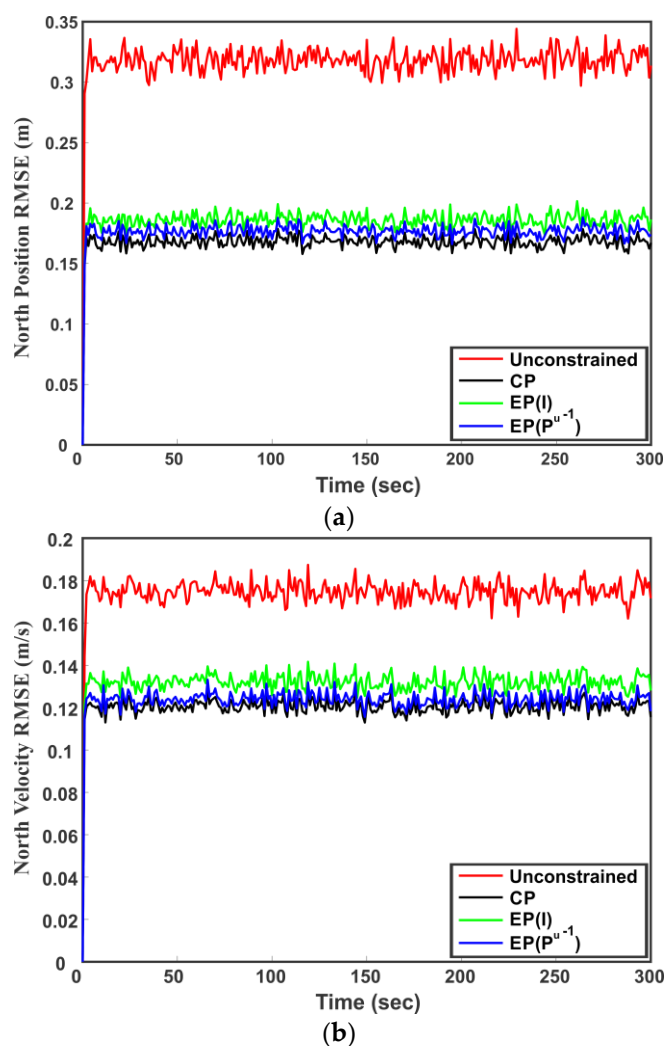


(**a**)



(**b**)

**Figure 10.** Simulation results for the constrained and unconstrained dynamic system. The covariance projection (CP) method is compared with the unconstrained state estimate and estimate projection (EP) method. (**a**) RMSE of northerly position of vehicle over 1000 runs; (**b**) RMSE of northerly velocity of vehicle over 1000 runs.

**Table 3.** Average RMSE for 1000 Monte Carlo Runs.

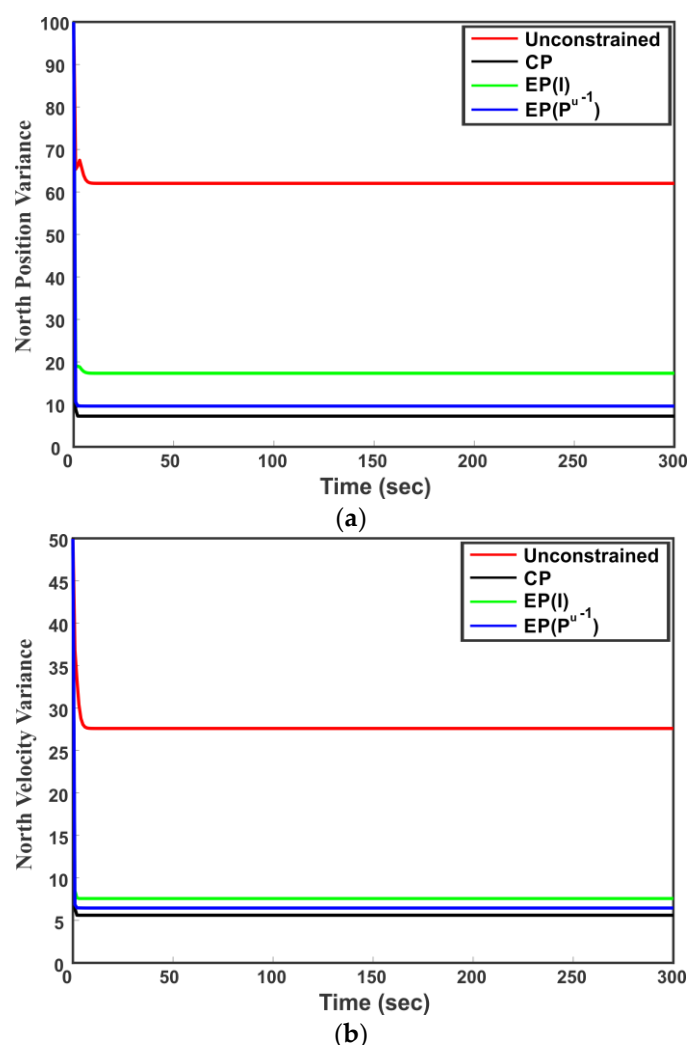| Methods | Average RMSE | | | |
|---|---|---|---|---|
| | $x_1$ (m) | $x_2$ (m) | $\dot{x}_1$ (m/s) | $\dot{x}_2$ (m/s) |
| Unconstrained | 95.479 | 87.496 | 52.413 | 49.972 |
| CP | 50.39 | 29.093 | 36.624 | 21.145 |
| EP (I) | 55.823 | 32.229 | 39.648 | 22.891 |
| EP ($P^{u-1}$) | 53.023 | 30.613 | 37.186 | 21.469 |

**Figure 11.** Comparative results of different methods in terms of states variance. (**a**) Variance of the northerly position of vehicle; (**b**) Variance of the northerly velocity of vehicle.

## 7. Conclusions

In this paper, we propose a general approach to data fusion under arbitrary correlations and linear constraints as well as data inconsistency. The proposed method provides an unbiased and optimal solution in the sense of MMSE for fusing data from multiple sources. The method improves the fusion accuracy by automatically detecting and removing inconsistent estimates from the fusion pool by a statistical confidence measure. Moreover, it is shown that by considering the cross-correlation among local estimates, the proposed method avoids the deterioration of the fusion accuracy due to Type I and Type II errors. Without any additional manipulation, the proposed method completely satisfies any linear constraints among state variables. The method improves the accuracy of constrained state estimation by satisfying the linear constraints among state variables and provides better results than the unconstrained state estimation and constrained state estimation using the estimate projection method.

Future work includes the extension of the proposed method for state estimation of non-linear dynamic systems. Another avenue is examining the proposed method for incorporating non-linear constraints among state variables.

**Author Contributions:** Sukhan Lee proposed the concept of data fusion based on covariance projection in the extended space as well as its application to outlier removal, while Muhammad Abu Bakr carried out the mathematical implementation and simulation analysis. Abu Bakr drafted the paper. Lee supervised Abu Bakr with critical assessment of the draft for quality revision.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

The fused mean estimate and covariance of the covariance projection (CP) method are given as,

$$\tilde{x} = W^{-1} P_r W \hat{x} \tag{A1}$$

$$\tilde{P} = W^{-1} P_r W P W^T P_r{}^T W^{-T} \tag{A2}$$

Putting $W = D^{-1/2} E^T$, $P_r = M^W \left( M^{W^T} M^W \right)^{-1} M^{W^T}$ and $M^W = WM$ in (A2), we get,

$$\tilde{P} = W^{-1}(WM(M^T W^T WM)^{-1} M^T W^T) \times (WM(M^T W^T WM)^{-1} M^T W^T)^T W^{-T}$$

Let $\alpha = M^T W^T WM$, then,

$$\tilde{P} = W^{-1} WM\alpha^{-1} M^T W^T WM\alpha^{-T} M^T W^T W^{-T}$$

$$\tilde{P} = M\alpha^{-T} M^T \tag{A3}$$

Putting the value of $\alpha$ in (A3) and simplifying, we get,

$$\tilde{P} = M(M^T E D^{-1} E^T M)^{-1} M^T \tag{A4}$$

By eigenvalue decomposition, we know that,

$$P = EDE^T \Rightarrow P^{-1} = ED^{-1}E^T$$

So,

$$\tilde{P} = M(M^T P^{-1} M)^{-1} M^T \tag{A5}$$

Similarly, using definitions of various components in CP fused mean (A1), we have,

$$\tilde{x} = W^{-1}(WM(M^T W^T WM)^{-1} M^T W^T)W\hat{x}$$

$$\tilde{x} = M(M^T W^T WM)^{-1} M^T W^T W\hat{x} \tag{A6}$$

Since $W^T W = P^{-1}$, (A6) can be simplified as,

$$\tilde{x} = M(M^T P^{-1} M)^{-1} M^T P^{-1} \hat{x} \tag{A7}$$

## Appendix B

The weighted distance from the joint mean of two data sources to the point on the manifold can be calculated as,

$$d = \begin{bmatrix} (\hat{x}_1 - \tilde{x})^T & (\hat{x}_2 - \tilde{x})^T \end{bmatrix} \begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix}^{-1} \begin{bmatrix} (\hat{x}_1 - \tilde{x}) \\ (\hat{x}_2 - \tilde{x}) \end{bmatrix}$$

$$d = (\hat{x}_1 - \tilde{x})^T P_1^{-1} (\hat{x}_1 - \tilde{x}) + (\hat{x}_2 - \tilde{x})^T P_2^{-1} (\hat{x}_2 - \tilde{x}) \tag{A8}$$

$$\hat{x}_1 - \tilde{x} = \hat{x}_1 - P_2(P_1 + P_2)^{-1}\hat{x}_1 + P_1(P_1 + P_2)^{-1}\hat{x}_2$$

$$\hat{x}_1 - \tilde{x} = [I - P_2(P_1 + P_2)^{-1}]\hat{x}_1 - [P_1(P_1 + P_2)^{-1}]\hat{x}_2$$

Since $P_1(P_1 + P_2)^{-1} + P_2(P_1 + P_2)^{-1} = I$

$$\hat{x}_1 - \tilde{x} = P_1(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2] \tag{A9}$$

Similarly,

$$\hat{x}_2 - \tilde{x} = [I - P_1(P_1 + P_2)^{-1}]\hat{x}_2 - [P_2(P_1 + P_2)^{-1}]\hat{x}_1$$

$$\hat{x}_2 - \tilde{x} = P_2(P_1 + P_2)^{-1}[\hat{x}_2 - \hat{x}_1]$$

$$\hat{x}_2 - \tilde{x} = -P_2(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2] \tag{A10}$$

Putting (A9) and (A10) in (A8) we get,

$$d = (P_1(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2])^T P_1^{-1}(P_1(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2])$$

$$+ (-P_2(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2])^T P_2^{-1}(-P_2(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2])$$

$$d = ([\hat{x}_1 - \hat{x}_2]^T (P_1 + P_2)^{-1} P_1) P_1^{-1}(P_1(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2])$$

$$+ ([\hat{x}_1 - \hat{x}_2]^T (P_1 + P_2)^{-1} P_2) P_2^{-1}(P_2(P_1 + P_2)^{-1}[\hat{x}_1 - \hat{x}_2])$$

$$d = [\hat{x}_1 - \hat{x}_2]^T [(P_1 + P_2)^{-1}(P_1 + P_2)(P_1 + P_2)^{-1}] [\hat{x}_1 - \hat{x}_2]$$

$$d = [\hat{x}_1 - \hat{x}_2]^T (P_1 + P_2)^{-1} [\hat{x}_1 - \hat{x}_2] \tag{A11}$$

## References

1. Liggins, M., II; Hall, D.; Llinas, J. *Handbook of Multisensor Data Fusion: Theory and Practice*; CRC Press: Boca Raton, FL, USA, 2017.
2. Hall, D.; Chong, C.; Llinas, J.; Liggins, M., II. *Distributed Data Fusion for Network-Centric Operations*; CRC Press: Boca Raton, FL, USA, 2012.
3. Kalman, R. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45.
4. Bar-Shalom, Y. On the track-to-track correlation problem. *IEEE Trans. Automat. Contr.* **1981**, *26*, 571–572.
5. Bakr, M.A.; Lee, S. Distributed Multisensor Data Fusion under Unknown Correlation and Data Inconsistency. *Sensors* **2017**, *17*, 2472, doi:10.3390/s17112472.
6. Maybeck, P. *Stochastic Models, Estimation, and Control*; Academic Press: Cambridge, MA, USA, 1982.
7. Bar-Shalom, Y.; Campo, L. The effect of the common process noise on the two-sensor fused-track covariance. *IEEE Trans. Aerosp.* **1986**, *AES-22*, 803–805.
8. Chang, K.C.; Saha, R.K; Bar-Shalom, Y. On optimal track-to-track fusion. *IEEE Trans. Aerosp. Electron. Syst.* **1997**, *33*, 1271–1276, doi:10.1109/7.625124.
9. Shin, V.; Lee, Y.; Choi, T. Generalized Millman's formula and its application for estimation problems. *Signal Process.* **2006**, *86*, 257–266.
10. Sun, S.; Deng, Z. Multi-sensor optimal information fusion Kalman filter. *Automatica* **2004**, *40*, 1017–1023.
11. Khaleghi, B.; Khamis, A.; Karray, F.; Razavi, S. Multisensor data fusion: A review of the state-of-the-art. *Inf. Fusion* **2013**, *14*, 28–44.
12. Abdulhafiz, W.; Khamis, A. Handling data uncertainty and inconsistency using multisensor data fusion. *Adv. Artif. Intell.* **2013**, *2013*, 11.
13. Durovic, Z.; Kovacevic, B. QQ-plot approach to robust Kalman filtering. *Int. J. Control* **1995**, *61*, 837–857.
14. Wellington, S.; Atkinson, J.; Sion, R. Sensor validation and fusion using the Nadaraya-Watson statistical estimator. In Proceedings of the Fifth International Conference on Information Fusion, Annapolis, MD, USA, 8–11 July 2002.
15. Hage, J.A.; Najjar, M.E.; Pomorski, D. Multi-sensor fusion approach with fault detection and exclusion based on the Kullback–Leibler Divergence: Application on collaborative multi-robot system. *Inf. Fusion* **2017**, *37*, 61–76.
16. Gobbo, D. Del; Napolitano, M.; Famouri, P.; Innocenti, M. Experimental application of extended Kalman filtering for sensor validation. *IEEE Trans. Control Syst. Technol.* **2001**, *9*, 376–380.
17. Brumback, B.; Srinath, M. A fault-tolerant multisensor navigation system design. *IEEE Trans. Aerosp. Electron. Syst.* **1987**, *AES-23*, 738–756.
18. Kumar, M.; Garg, D.; Zachery, R. A method for judicious fusion of inconsistent multiple sensor data. *IEEE Sens. J.* **2007**, *7*, 723–733.

19. Kumar, M.; Garg, D.; Zachery, R. A generalized approach for inconsistency detection in data fusion from multiple sensors. In Proceedings of the 2006 American Control Conference, Minneapolis, MN, USA, 14–16 June 2006.

20. Uhlmann, J. Covariance consistency methods for fault-tolerant distributed data fusion. *Inf. Fusion* **2003**, *4*, 201–215.

21. Kirubarajan, T.; Bar-Shalom, Y.; Pattipati, K.; Kadar, I. Ground target tracking with variable structure IMM estimator. *IEEE Trans. Aerosp. Electron. Syst.* **2000**, *36*, 26–46.

22. Bernstein, D.S.; Hyland, D.C. Compartmental Modeling and Second-Moment Analysis of State Space Systems. *SIAM J. Matrix Anal. Appl.* **1993**, *14*, 880–901, doi:10.1137/0614060.

23. Simon, D. Kalman filtering with state constraints: a survey of linear and nonlinear algorithms. *IET Control Theory Appl.* **2010**, *4*, 1303–1318, doi:10.1049/iet-cta.2009.0032.

24. Simon, D.; Chia, T. Kalman filtering with state equality constraints. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 128–136.

25. Wen, W.; Durrant-Whyte, H. Model-based multi-sensor data fusion. In Proceedings of the 1992 IEEE International Conference on Robotics Automation, Nice, France, 12–14 May 1992.

26. Tahk, M.; Speyer, J. Target tracking problems subject to kinematic constraints. *IEEE Trans. Automat. Contr.* **1990**, *35*, 324–326.

27. Porrill, J. Optimal Combination and Constraints for Geometrical Sensor Data. *Int. J. Robot. Res.* **1988**, *7*, 66–77, doi:10.1177/027836498800700606.

28. Geeter, J. De; Brussel, H. Van; Schutter, J. De; Decreton, M. A smoothly constrained Kalman filter. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 1171–1177.

29. Hewett, R.J.; Heath, M.T.; Butala, M.D.; Kamalabadi, F. A Robust Null Space Method for Linear Equality Constrained State Estimation. *IEEE Trans. Signal Process.* **2010**, *58*, 3961–3971, doi:10.1109/TSP.2010.2048901.

30. Bakr, M.A.; Lee, S. A general framework for data fusion and outlier removal in distributed sensor networks. In Proceedings of the 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Daegu, Korea, 16–18 November 2017.

31. Simon, D. *Optimal State Estimation: Kalman, H [Infinity] and Nonlinear Approaches*; Wiley-Interscience: Hoboken, NJ, USA, 2006; ISBN 0470045337.

32. Jiang, L. Sensor Fault Detection and Isolation Using System Dynamics Identification Techniques. Ph.D. Thesis, University of Michigan, Ann Arbor, MI, USA, 2011.

33. Bar-Shalom, Y.; Willett, P.; Tian, X. *Tracking and Data Fusion*; YBS publishing: Storrs, CT, USA, 2011.

34. Walpole, R.; Myers, R.; Myers, S.; Ye, K. *Probability and Statistics for Engineers and Scientists*; Macmillan: Basingstoke, UK, 1993.

35. Fishman, G.S. *Monte Carlo*; Springer: New York, NY, USA, 1996; ISBN 978-1-4419-2847-4.

36. Rillo, G.; Morales, M.A.; Ceperley, D.M.; Pierleoni, C. Coupled electron-ion Monte Carlo simulation of hydrogen molecular crystals. *J. Chem. Phys.* **2018**, *148*, 102314, doi:10.1063/1.5001387.

37. Baudry, G.; Macharis, C.; Vallée, T. Range-based Multi-Actor Multi-Criteria Analysis: A combined method of Multi-Actor Multi-Criteria Analysis and Monte Carlo simulation to support participatory decision making under uncertainty. *Eur. J. Oper. Res.* **2017**, *264*, 257–269, doi:10.1016/j.ejor.2017.06.036.

38. Ma, Y.; Chen, X.; Biegler, L.T. Monte-Carlo-simulation-based optimization for copolymerization processes with embedded chemical composition distribution. *Comput. Chem. Eng.* **2018**, *109*, 261–275, doi:10.1016/J.COMPCHEMENG.2017.11.018.

39. Vîlcu, A.-D.; Vîlcu, G.-E. An algorithm to estimate the vertices of a tetrahedron from uniform random points inside. *Ann. Mat. Pura Appl.* **2018**, *197*, 487–500, doi:10.1007/s10231-017-0688-6.

40. Hua, X.; Cheng, Y.; Wang, H.; Qin, Y. Robust Covariance Estimators Based on Information Divergences and Riemannian Manifold. *Entropy* **2018**, *20*, 219, doi:10.3390/e20040219.

41. Ljung, L. System Identification. In *Signal Analysis and Prediction*; Prochazka, A., Kinsbury, N., Payner, P.J.W., Uhlir, L., Eds.; Birkhäuser: Boston, MA, USA, 1998.